

# **The Effects of Construct Redundancy on Readers' Understanding of Conceptual Models**

Palash Bera  
Operations and IT Management  
John Cook School of Business  
3674 Lindell Blvd.  
Saint Louis University  
St. Louis, MO, 63108, USA  
Ph. 314-977-3867  
Email: [pbera@slu.edu](mailto:pbera@slu.edu)

Geert Poels  
Faculty of Economics and Business Administration  
Department of Business Informatics and Operation Management  
University of Ghent  
Twekerkenstraat 2  
9000 Gent, Belgium  
Ph. 09-264-34 97  
Email: [Geert.Poels@UGent.be](mailto:Geert.Poels@UGent.be)

## **Abstract**

This paper investigates the effect of construct redundancy on readers' understanding of conceptual models. Conceptual models play a crucial role in understanding the domain related to information system development. The clarity of such models can be compromised if they are constructed using a conceptual modelling grammar exhibiting construct redundancy where one real-world phenomenon maps to two or more grammar constructs. With two empirical studies on solving domain-related problems using Unified Modeling Language (UML) class diagrams as conceptual domain models, it was found that when construct redundancy is present at different strengths, then the effect of the redundancy on the understanding of a model depends on the modeling knowledge of the reader. Novice readers with minimal modeling knowledge find models difficult to interpret when a strong level of redundancy caused by distinct construct redundancy exists. However, when the models have a weak level of redundancy then these readers find them easier to understand compared to models without redundancy. In contrast, trained readers are indifferent to a weak level of redundancy in a model.

## **1. Introduction**

Information systems (IS) are representations of other real-world systems. Wand and Weber (2017) suggest that to determine what constitutes a good representation, we need to refer to ontological theories. They introduced a theory of ontological expressiveness (1993) that predicts that if a conceptual modeling grammar does not have a one-one mapping between its constructs and the constructs of an ontological theory then the scripts generated by the grammar will not be clean and complete. One way to compromise ontological expressiveness is if an ontological construct that conceptualizes some real-world phenomenon maps to two or more grammatical constructs, a situation known as construct redundancy. Such a compromise might reduce the clarity of a model and thus affect its interpretation.

The literature has identified construct redundancy in conceptual modeling grammars such as in UML (Opdahl & Henderson-Sellers, 2002) and in process modeling languages (Recker, Indulska, Rosemann, & Green, 2010). However, the empirical research in this area is limited. Fickinger and Recker (2013) find that the studies on redundancies in process modeling obtain mixed results. Green and Rosemann (2001) find that construct redundancy in the views of the ARIS framework provides users a mechanism to manage complexity in modeling. Recker et al. (2010) find that practitioners have little problems with construct redundancy in the BPMN modelling grammar. Modelers with medium to extensive experience seemed to be more aware of ontological deficiencies in BPMN (including instances of construct redundancy), but could mitigate these deficiencies by developing workarounds. Recker and Rosemann (2010) find that experienced modelers perceive ontological deficiencies in BPMN (including construct redundancy related to the representation in BPMN diagrams of events and transformations) more strongly than inexperienced modelers. Recker et al. (2010) find that construct redundancy in the BPMN representation of real-world objects and events negatively impacts the perceived ease of use of BPMN but that this effect only holds for users that were previously confronted with modeling situations where ambiguities related to construct redundancy showed up. Thus, the effect of construct redundancy on model understanding might depend on the readers' knowledge of the models. In the same vein, Fickinger and Recker (2013) not only suggest that more empirical research should be carried out on construct redundancy but such research should focus on individual differences. Accordingly, in this paper, we focus on the question of *how conceptual models with construct redundancy affect the understanding of readers with varied modeling knowledge*.

In section 2, we discuss ontological expressiveness and operationalize two kinds of construct redundancy with different strengths which we illustrate using UML class diagrams. Using cognitive and learning theories, we predict that these construct redundant models have different effects on the readers' understanding based on their modeling knowledge. We test the prediction in section 3 by conducting two empirical studies. We discuss our results in section 4 and present a conclusion in section 5.

## **2. Theoretical Background and Prediction**

Ontological theory explains the structure and behavior of the world. If one agrees to the worldview expressed by the ontological theory, then a mental model of some real-world situation can be created in terms of the ontological constructs and this model can be externalized by articulating it with grammatical constructs (Guizzardi, 2005). The theory of ontological expressiveness (Wand & Weber, 1993) identifies four types of ontological deficiency of a conceptual modeling grammar that might affect the ability to represent some real-world phenomena completely and clearly. These deficiencies are: (a) *construct deficit* – when the grammar contains no construct that maps to a particular ontological construct, (b) *construct excess*- when the grammar contains a construct that does not map to any ontological construct, (c) *construct overload*- when the grammar contains a construct that maps to two or more ontological constructs, and (d) *construct redundancy*- when the grammar contains two or more constructs that map to a single ontological construct. Wand and Weber (2017) mention that construct deficit undermines the ontological completeness of a grammar and ontological excess, overload, and redundancy undermine its ontological clarity.

As the theory of ontological expressiveness can by itself not explain why a lack of ontological clarity impacts the ability to understand a conceptual modeling script, researchers

have resorted to various theories of perception, cognition, memory, and language to propose empirically verifiable hypotheses. Using such theories, several conceptual modeling grammars or some of their constructs have been evaluated. This includes the use of entities to represent both things and events (*construct overload*) (Allen & March, 2006), the use of optional attributes and relationships to represent optional properties (*construct excess*) (Bodart, Patel, Sim, & Weber, 2004; Bowen, O'Farrell, & Rohde, 2009; Dunn, Gerard, & Grabski, 2011; Gemino & Wand, 2005), the use of relationships or associations to represent composites (*construct overload*) (Graeme Shanks, Tansley, Nuredini, Tobin, & Weber, 2008), and the use of entities to represent intrinsic properties (*construct overload*) (Weber & Zhang, 1996). Recently, Tilakaratna & Rajapakse (2017) evaluated the ontological completeness and clarity of UML and suggest a subset of UML constructs that is better suitable for conceptual modeling.

The types of deficiencies regarding ontological clarity investigated in these studies are mostly construct overload and construct excess. The lack of studies investigating construct redundancy is surprising as numerous ontological deficiencies of the construct redundancy type have been discovered in conceptual modeling grammars like UML (Opdahl & Henderson-Sellers, 2002) and process modeling languages (Recker et al., 2010). As stated in the introduction, we noted that empirical studies investigating predictions derived from ontological analysis of process modeling languages found mixed effects (i.e., positive, neutral or negative) related to construct redundancy. Furthermore, these effects might depend on user's knowledge of or experience with the grammar (Fickinger & Recker, 2013).

In the first sub-section, we further explore the concept of construct redundancy, discuss different types that result in weak or stronger forms of redundancy, and illustrate these using a grammar based on UML class diagrams. Next, we review relevant theories on which predictions

regarding the effect of weak and strong construct redundancy on model understanding can be built, differentiating between novice and more advanced model users.


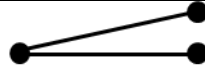
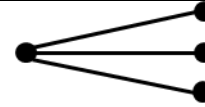
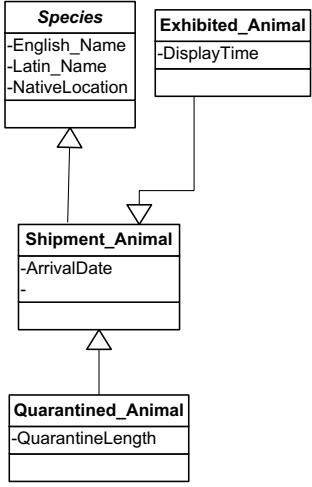
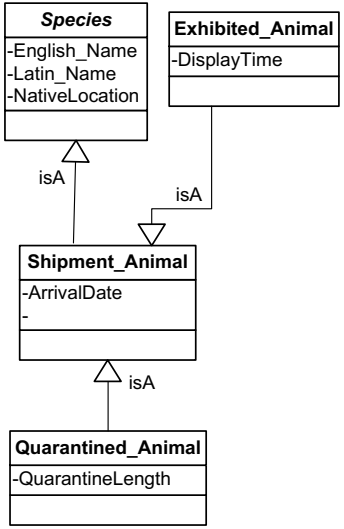
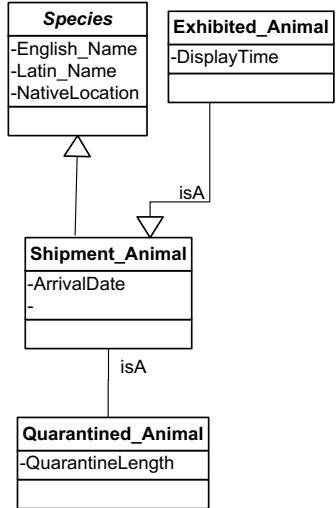
## 2.1 Construct Redundancy in UML Class Diagrams

Conceptual modeling grammars such as the UML and ERD are often used to formally describe some aspects of the domain for the purpose of understanding and communicating about it (Mylopoulos, 1992). The UML class diagram is a popular conceptual model in practice (Dobing & Parsons, 2006). Construct redundancy can confuse the users of a conceptual model because the use of multiple grammatical constructs to represent some type of real-world phenomenon creates ambiguity in the interpretation of the phenomenon (Weber, 2003). To illustrate this, we focus on three ontological constructs, *generalization*, *composition*, and *aggregation*, as they are commonly represented in UML class diagrams. We introduce redundancy at two levels to create *construct redundant UML class diagrams*. At the first level, we introduce redundancy in the form of labels. Labels are added to the current symbols of generalization, composition, and aggregation. Thus, in the class diagrams, the label “is part of” is redundant in representing the aggregation construct. Similarly, the labels “isComponentOf” and “isA” are redundant for the constructs composition and generalization respectively. We call these class diagrams *weak construct redundant UML class diagrams*. At the second level, we use three alternative (combinations of) grammatical constructs to represent an ontological construct. For instance, to represent generalization, we use either the UML generalization symbol, a combination of the UML generalization symbol and the “isA” label (as in weak construct redundant UML class diagrams), or the combination of an UML association line and the “isA” label. Similarly, redundancy is introduced in aggregation and composition constructs. These class diagrams are termed *strong construct redundant UML class diagrams*. Whereas in weak construct redundant

UML class diagrams, redundancy is only manifested in the grammatical construct (i.e., the label is redundant), in strong construct redundant UML class diagrams real-world phenomena corresponding to a same ontological construct (e.g., generalization) are represented by different (combinations of) grammatical constructs in the same model. This latter form of redundancy is similar to the concepts of *distinct construct redundancy* of Fickinger and Recker (2013) and *non-laconicity* of Guizzardi (2005), and is considered to be a detrimental form of construct redundancy, hence it's labelled as 'strong.'

Table 1 illustrates the two types of construct redundant UML class diagrams along with a corresponding ontologically clear UML class diagram (i.e., no presence of weak or strong construct redundancy). Note that the weak construct redundant model exhibits the property of *laconicity* (Guizzardi, 2005), meaning that although the grammar is characterized by the ontological deficiency of construct redundancy, the model constructed with the grammar is consistent throughout the model (i.e., all generalizations are represented using the combination of UML generalization symbol and "isA" label). In the strong redundant UML class diagrams, multiple grammatical constructs in the same model create complexity that can hamper the model's understandability. The exact meaning of generalization is difficult to interpret when it is represented in three different ways. Thus, a question such as "is a quarantined animal a different type of generalization than an exhibited animal?" can arise in the readers' mind. Such redundant class diagrams can become even more complex when composition and aggregation are also represented in multiple ways.

Table 1: Examples of ontologically clear and weak and strong construct redundant models

Clear model	Weak redundant model	Strong redundant model
 <p>Ontological construct      Grammatical construct</p>	 <p>Ontological construct      Grammatical construct</p>	 <p>Ontological construct      Grammatical construct</p>
		
Grammatical construct UML generalization symbol represents the ontological construct generalization	Grammatical construct “isA”-labelled UML generalization symbol represents the ontological construct generalization. The information conveyed by the “isA” label is redundant to that of the generalization symbol	The ontological construct generalization is represented in three different ways: UML generalization symbol; “isA”-labelled UML generalization symbol; “isA”-labelled UML association symbol.

## 2.2 Effects of Readers’ Knowledge on Understanding Construct Redundant Models

We propose that the readers’ knowledge of modeling techniques might affect their understanding of models. Accordingly, we discuss the differences between expert and novice readers and then we discuss theories on how the readers’ model understanding is affected by weak and strong construct redundant models.

Klayuga et al. (2003) differentiate experts from novices. While experts possess domain specific schemas that help them to categorize information at higher level and solve problems,



novices do not and therefore require greater cognitive effort to solve problems. If the effort to solve a problem is large, then chances increase that cognitive overload is imposed on the novice (Ward & Sweller, 1990). Research shows that experts possess an extensive long-term memory with structured information (Glaser, 1984). Thus for solving problems, experts often bypass their working memory and rely on their structured information that then reduces the burden on their working memory (Chandler & Sweller, 1991). In contrast, novices, with limited acquired knowledge, have greater difficulty in solving problems as they need to process information through a limited working memory. Similarly, Mayer (1989) points out that those learners' who have low prior knowledge have difficulty in creating mental representations of tasks.

In the context of conceptual modeling, several researchers have investigated the difference between novice and expert analysts. Batra and Davis (1992) investigate the performance of experts and novices in developing conceptual models. They find that experts are able to conceptualize and understand model descriptions better than novices. Shanks (1997) traces the cognitive process of expert and novice data modelers and finds that expert modelers are better able to conceptualize and understand a case's description than novices. In terms of the quality of the models developed by the experts, Shanks (1997) finds that data models are more correct, complete, innovative, flexible, and better understood than those built by novices.

Theories related to cognitive psychology can be used to predict the effect of strong and weak redundant UML class diagrams on novice and expert readers. The theories on secondary notation and the redundancy effect are particularly relevant to novice readers. Secondary notation (Petre, 2006) is the use of symbols, graphs, or texts in a conceptual model to denote elements in real-world situations that are redundant to the grammatical constructs. Secondary notation reduces the cognitive effort involved in interpreting conceptual modeling scripts

(Moody, 2009). The “isA”, “isComponentOf”, and “isPartOf” labels used in the operationalization of weak construct redundant UML class diagrams (see middle column, Table 1) are examples of such a notation in conjunction with the UML generalization, composition and aggregation symbols. These labels do not convey any additional information about the relationship but are used as a clue (Petre, 2006) that speeds up the perception and subsequent recognition of the relationship semantics. Moody (2009) stresses in the dual coding principle of the Physics of Notations theory that the label has a reinforcing function and should not replace the symbolic grammatical construct.

The redundancy effect relates to how redundant elements in a diagram (e.g., text and figures) affect the reader’s ability to learn (Chandler & Sweller, 1991). Chandler and Sweller propose that if readers can fully learn from a diagram alone, then the text in the diagram becomes redundant. Integrating the diagram with the text is thus likely to unnecessarily force readers to process the text that leads to negative rather than positive effects. They also find that if redundant elements can easily and readily be assimilated by the readers, then the inclusion of such elements might not have any negative consequences. The redundancy effect could also be complementary if learners are not able to fully learn from a diagram (Ainsworth, 2006) and need the additional information. This need might occur when learners are not familiar with the constructs in the diagrams. In this circumstance, when learners find it difficult to understand the structure of the diagrams, they might find an intermediate representation easier to understand (Robertson, 2003). Kalyuga et al. (2003) report several studies that show that less knowledgeable learners benefit from redundant text but for those with more knowledge, the added text interferes with their learning from the diagram.

For novice readers, the redundancy in weak construct redundant models aids in understanding as the labels act as complementary elements in understanding unfamiliar symbols. However, expert readers might ignore the labels in the weak construct redundant models as they are already familiar with the grammatical constructs.

When novice readers use strong construct redundant models, they have a higher chance to experience cognitive overload (Sweller, 1988) as humans have limited cognitive resources that are available during learning and problem solving. When the same ontological construct is displayed in using multiple grammatical constructs, novices get confused and their cognitive load increases. In this case, grammatical constructs no longer have a complementary effect. We expect that expert readers will be less confused by the use of multiple grammatical constructs in the strong construct redundant models, hence there will be less chance of cognitive overload. Table 2 summarizes the above discussion by showing the effects of model understanding based on model and reader types.

Table 2: Effects of model types on model understanding

Model type	Reader type	Effect on model understanding	Supported theory
Weak construct redundant	Novice	Redundancy aids in model understanding	Secondary notation and complimentary redundancy effect
Strong construct redundant	Novice	Redundancy does not aid but rather creates confusion in model understanding	Cognitive overload
Weak construct redundant	Expert	Redundancy does not aid in model understanding	No redundancy effect
Strong construct redundant	Expert	Redundancy does not aid in model understanding	No redundancy effect

Based on Table 2, we propose that the effects of construct redundancy (i.e., beneficial, neutral, and detrimental) depend on the reader's modeling knowledge. If the readers are well-versed in

the modeling language, then manipulating redundancy will not have any effect on their model understanding. However, if the readers are not well-versed in the modeling language, then the effect of weak construct redundant models on model understanding will be beneficial but the effect of strong construct redundant models on model understanding will be detrimental. Figure 1 illustrates these proposals. The horizontal blue line represents expert readers and the red line represents novice readers. The Y axis represents understanding performance as a dependent variable and the combinations of model types in the X axis (e.g., clear versus weak redundant versus strong redundant) represent the independent variable. An assumption is made here that expert readers will overall perform better than novice readers. However, the actual performance on model understanding (i.e., high, medium, or low) will depend on the subjects who are selected as expert and novice readers.

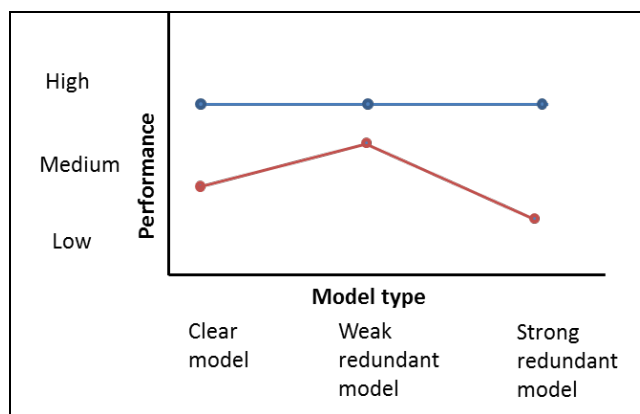


Figure 1: The effect of readers' modeling knowledge and model type on task performance

### 3. Empirical Studies

In the first study we test the ontological prediction captured in Figure 1 (sub-section 3.1). The aim of the second study was to obtain a better understanding of the underlying mechanism of the observed effects (sub-section 3.2).

#### 3.1 Empirical Study 1

##### *Tasks*

In this study, we use problem-solving tasks as a measure of the understanding of conceptual models. Problem solving is a cognitive process where a problem solver finds “a way out of a difficulty, a way around an obstacle, attaining an aim that was not immediately attainable” (Polya, (1968 p. ix). Understanding a domain is equivalent to learning. The outcomes of learning can be evaluated as a transfer (Mayer, 1989) in which the ability to use knowledge gained from the material is used to solve related problems not directly answerable from the material (Gemino & Wand, 2005). Accordingly, a high score on problem-solving questions reflects a high understanding of the model (Gemino, 1998; Mayer, 1989).

We develop three sets of UML class diagrams as per the three variations (clear, weak construct redundant, and strong construct redundant). To increase external validity, each set has two class diagrams: “flight management” and “aquarium management”. Thus, we prepare in total six class diagrams. Appendix A shows all the variations of these models.

The clear class diagrams use only the conventional UML symbols for generalization, aggregation, and composition. The weak construct redundant diagrams use the labelled symbols for these relationships. In the strong construct redundant diagrams, three alternative (combinations of) grammatical constructs are used to represent different instances of the same ontological construct. However, as there are less than three generalization, composition and aggregation constructs in each diagram, we cannot include all three types of grammatical constructs for the same ontological construct in the same diagram. Rather we spread them across two diagrams. For example, in the flight model, aggregation is represented by “symbol and label” and by “association line and label,” and in the aquarium model, aggregation is represented by symbol only. We develop three problem-solving questions for each model. Because the

answers for these questions are open-ended, we use two graders. The responses are graded as correct and incorrect.

We now demonstrate how expert readers and novice readers answer a problem-solving question. The domain used for this purpose is “flight management.” Three fragments of conceptual models from this domain are used (shown in Table 3) where each fragment represents an ontologically clear, weak construct redundant, or a strong construct redundant model. Table 3 provides reasons for why novice readers may find weak redundant models more useful than clear models and clear models more useful than strong redundant models. The table also provides justification on why expert readers do not prefer any particular model over others.

Table 3: Illustrating the effects of the three types of UML models on problem-solving questions

<p><b>Sample problem-solving question:</b> <i>Some of the safety procedures were not followed by a few crew members during a certain chartered flight. Explain how this could have happened?</i></p>		
<p><b>Fig 2A:</b> Example of a fragment of a clear model.</p>	<p><b>Fig 2B:</b> Example of a fragment of a weak redundant model.</p>	<p><b>Fig 2C:</b> Example of a fragment of a strong redundant model.</p>
	<b>Novice Readers</b>	<b>Expert Readers</b>
<b>Relevant correct responses to the problem-solving question</b>	<b>Why 2B should be more effective than 2A and 2A more effective than 2C in identifying such responses?</b>	<b>Why there should not be any difference in using 2A, 2B, or 2C in identifying such responses?</b>

1. The flight engineers were not trained adequately. 2. Some of the crew members other than flight engineers were absent on that flight.	Better understanding of the relationship among crew, pilot, and flight engineer can lead to a higher number of correct responses. Referring to Fig 2A, novice readers are not much familiar with the aggregation symbol between the flight engineer and crew. But the meaning of aggregation is clear in Fig 2B. In Fig 2C, aggregation is represented by a new symbol hindering the understanding of the concepts in this figure. Also presence of multiple symbols (in other parts of the model) to represent aggregation should confuse the novice readers. Thus, novice readers should understand Fig 2B better than Fig 2A and Fig 2A better than Fig. 2C.	The presence of the label “isPartOf” in Fig. 2B does not have any effect on the expert readers’ understanding of the model as they are already familiar with the aggregation symbol. On the other hand, the cardinality of one in the new aggregation symbol in Fig 2C provides a clue to the expert readers on how to read the direction of the aggregation that is linked with the classes. In both cases (Fig. 2B and 2C), expert readers’ understanding of the models does not change when compared to the understanding of Fig. 2A.
---	---	--

### ***Hypotheses***

To understand the effect of the redundancies in models, they should be compared to the ontological clear model as a benchmark. We develop hypotheses based on the information provided in Tables 2 and 3.

We operationalize expert readers by using graduate students who have substantial knowledge of UML class diagrams. We name such subjects as *trained readers*. We use undergraduate students who had minimum knowledge on UML class diagrams as *novice readers*. The details of the subjects are provided in the next section. To ensure that the desired effects (as indicated in Table 3) are obtained, undergraduate students are deliberately chosen as novice readers. With bare minimum knowledge of the constructs of generalization, composition, and aggregation, it is expected that there will be a high chance of cognitive overload on these readers when they refer to the strong redundant models. With up to nine grammatical constructs

representing the three ontological constructs, the novices are expected to get confused on the real interpretations of the grammatical constructs.

We expect that for novice readers, the redundancy effect becomes complementary when they refer to the weak redundant models, whereas for trained readers, the redundancy effect will not be manifested when they refer to the redundant models. Sweller (1988) finds that redundancy effects are likely to be unnoticed when the intrinsic (i.e., complexity of the models) cognitive load is low. For trained readers, low intrinsic load is not likely to result in cognitive overload because of their greater knowledge of the constructs in the models. Accordingly, the hypotheses for the study are the following:

*H1: Novice readers will obtain a better understanding of the domain when reading weak redundant UML class diagrams compared to reading clear UML class diagrams.*

*H2: Novice readers will obtain a better understanding of the domain reading ontologically clear UML class diagrams compared to reading strong redundant UML class diagrams.*

*H3: Trained readers do not have a significantly better understanding of the domain by reading ontologically clear UML class diagrams compared to reading weak redundant UML class diagrams.*

*H4: Trained readers do not have a significantly better understanding of the domain by reading ontologically clear UML class diagrams compared to reading strong redundant UML class diagrams.*

The dependent variable is the number of correct responses to the problem-solving questions. The independent variable is the use of different types of the class diagrams as mentioned in the hypotheses. To test the above hypotheses, we conduct a laboratory study with student subjects, which is common in conceptual modeling studies (Burton-Jones & Meso, 2006).



### ***Experiment design***

The study has two factors (trained and novice readers) in each type of UML class diagram—clear, weak redundant, and strong redundant. Thus, we designed a between-subjects study with six groups where each subject receives only one type of diagram. For this purpose, three sets of class diagrams were developed that are similar to the diagrams in Table 3. To increase the external validity, each set had class diagrams developed in two models: “flight management” and “aquarium management.”

### ***Subjects and procedure***

The study had 120 subject participants in four groups of 20 and 21 for the group where novice readers received clear models and 19 for the group where trained subjects received strong redundant models. A total of 110 subjects (18 in each group) are required to achieve 80% power at two sided 5% significance level. Undergraduate and graduate students enrolled in a MIS program from a Southern US University were recruited as subjects. A key issue in this experiment was to recruit subjects so that they represented novice and trained readers. To ensure this, subjects as trained readers were recruited from MIS graduate students who were finishing the IS design courses. Prior to participation, through course work, these subjects were trained extensively in the use of UML class diagrams. These subjects also learned the UML concepts from the IS analysis course that they took prior to the IS design course. Thus, the subjects were exposed to UML concepts over a period of two semesters. As experts depend on their long-term memory to perform tasks (Glaser, 1984), some length of exposure is required to get expertise in UML class diagrams. For novices, subjects were enrolled in undergraduate introductory IS and IS analysis courses. In these courses, subjects were taught about the modeling concepts (primarily through ER diagrams) but were not exposed to UML class diagrams prior to the study.

Prior to participation, the novice readers were trained for three hours on the basics of a UML class diagram - classes, attributes, operations, and associations. These subjects were not exposed to generalization, composition, and aggregation concepts in UML. This was done to get the maximum effect of the treatment because prior exposure to the constructs would bias them and would have threatened the internal validity of the research. Each subject was randomly allocated to one of the three types of UML class diagrams (clear, weak redundant, and strong redundant).

The following experimental procedure was followed: First, subjects answered background questions on their modeling knowledge (also referred to as IS domain knowledge (Khatri, Vessey, Ramesh, Clay, & Park, 2006)). To check the level of UML knowledge possessed by these subjects, they were asked to draw a UML class diagram that involved modeling classes and associations. Following this step, all subjects browsed through the general concepts of a UML class diagram that focused on the three ontological concepts of generalization, composition, and aggregation. These constructs were shown using the usual UML symbols (i.e., triangle, black diamond, and white diamond). Thus, subjects who were placed in the weak and strong redundant groups were unaware of the new grammatical constructs (i.e., labels, labels and association line) that were mentioned in the experimental class diagrams. Next, subjects were trained to answer problem-solving tasks. This was important as answering problem-solving questions requires the assimilation of information from the diagram and the subjects' prior knowledge. Subjects then answered a set of comprehension questions (True or False) that covered the content of the diagram. The purpose of this task was to familiarize and involve the subjects with the diagram. Following this step, the subjects answered three problem-solving questions. This sequence of answering comprehension and problem-solving questions

was repeated for another domain from a different model but in the same experimental condition.

The order of the diagrams provided to the subjects was reversed for half of the subjects. This was done to cancel any domain order effect. The experimental materials are provided in Appendix B.

## Results

The UML class diagrams developed by subjects were marked by an independent coder who coded each diagram on a scale of 1 to 5 (1, very poor representation and 5, excellent representation). The average scores for the novice groups using the clear and weak and strong redundant diagrams respectively were 3.48, 3.60, and 3.50; and for the trained groups, they were 4.25, 4.40, and 4.37. A t-test analysis ( $t\text{-test} = 6.31$ ,  $p = 0.00$ ) for the mean score difference established the difference in UML proficiency between the two groups (novice and trained).

Table 4 provides the descriptive statistics for the study.

Table 4: Descriptive Statistics

Variables	Scale	C M	C SD	RW M	RW SD	RS M	RS SD	C M	C SD	RW M	RW SD	RS M	RS SD	Total M	Total SD
		N=20		N=20		N= 19		N=21		N=20		N=20		N=120	
		Trained Readers						Novice Readers							
Flight fam.	1-7	3.58	0.37	3.73	0.41	3.74	0.45	3.40	0.58	3.55	0.48	3.62	0.42	3.60	0.46
Aquarium fam.	1-7	3.78	0.37	3.85	0.33	3.74	0.31	3.83	0.72	3.65	0.49	3.77	0.34	3.77	0.45
Domain fam.	1-7	3.68	0.27	3.79	0.22	3.74	0.26	3.62	0.53	3.60	0.43	3.70	0.24	3.69	0.34
Modeling Kno.	1-7	4.45	0.35	4.52	0.43	4.41	0.30	3.60	0.39	3.63	0.38	3.44	0.37	4.00	0.58
PS-flight	1-5*	2.70	0.76	2.53	0.93	2.21	0.42	1.49	0.43	1.78	0.73	1.03	0.36	1.97	0.86
PS-aquarium	1-4*	2.63	0.83	2.78	0.83	2.23	0.64	1.56	0.32	2.15	0.96	1.23	0.41	2.09	0.88
Problem solving	1-4*	2.67	0.65	2.66	0.80	2.22	0.36	1.52	0.30	1.97	0.79	1.13	0.27	2.03	0.79

**Key:** Comp: Average number of correct comprehension score; PS: Average number of correct problem-solving responses per question; fam-familiarity; Kno.- Knowledge, C- Clear; RW- Redundant-Weak; RS- Redundant-Strong; M- Mean; SD- Standard Deviation; \*\*The problem-solving questions were open-ended, so the maximum score is undefined. However, the list of correct answers suggests a “practical” maximum of 5 and 4 for the two domains

The scores on domain familiarity of the trained readers are similar to those of the novices across the three treatments. However, the scores on modeling knowledge of the trained readers are

higher than the novices across the three treatments. This difference matches with the earlier reported manipulation test on the subject's proficiency in creating UML class diagrams.

The reliability of the instruments for the dependent variable, problem solving is checked next. Cronbach's alpha for items measuring the dependent variable is 0.83 (aquarium model) and 0.88 (flight model). The two graders evaluated the responses of the problem-solving questions using a code book that contained sample answers for each question. Grader one coded all the responses, and grader two coded only one-third of the responses randomly. Their inter-rater reliability (alpha) is high (average Pearson correlation coefficient for the correct number of responses is 0.89).

Tables 5 and 6 show the ANCOVA results for each domain with the effects of the covariates for hypotheses 1-2 and 3-4 respectively. To test the above hypotheses, the average number of correct responses of the problem solving questions for both domains was used as the measure for dependent variable.

Table 5: ANCOVA Analysis H1-H2 (novice readers)

ANCOVA analysis for novice readers	H1 (C-RW difference)		H2 (C-RS difference)	
	PS RW (1.97) > PS C (1.52)		PS C (1.52) > PS RS (1.13)	
Variables	F	P	F	P
Order	0.70	0.21	0.43	0.26
Domain familiarity	0.95	0.17	0.46	0.25
Treatment	7.24	0.01*	8.67	0.01*
	<i>Partial eta squared = 0.17</i>		<i>Partial eta squared = 0.19</i>	

**Key:** Order- Order of the domains presented to the subjects, PS- Problem solving average correct score C- Clear, RW- Redundant Weak, RS- Redundant Strong, \*Significance at 0.05 level

Table 6: ANCOVA Analysis H3-H4 (trained readers)

ANCOVA analysis for trained readers	H3 (C-RW difference)		H4 (C-RS difference)	
	PS C (2.67) > PS RW (2.66)		PS C (2.67) > PS RS (2.22)	
Variables	F	P	F	P
Order	0.04	0.42	0.01	0.48
Domain familiarity	1.75	0.10	0.01	0.47
Treatment	0.02	0.45	3.22	0.04*

	<i>Partial eta squared = 0.01</i>	<i>Partial eta squared = 0.08</i>
--	-----------------------------------	-----------------------------------

**Key:** Order- Order of the domains presented to the subjects, PS- Problem solving average correct score C-Clear, RW- Redundant Weak, RS- Redundant Strong, \*Significance at 0.05 level

The average correct problem-solving scores for trained readers are higher than for the novices.

Tables 5 and 6 indicate that H1, H2, and H3 were supported but not H4. The results are consistent with the prediction on how novices and trained readers react to weak and strong redundant models. As H4 was not supported therefore it might be possible that trained readers faced cognitive overload using the strong redundant models. The partial eta squared for H4 analysis is low (0.08) indicating a very small effect size.

### **3.2 Empirical Study 2**

#### ***Background***

From study 1 there is evidence that trained readers can better understand UML class diagrams than novice readers. However, there is no evidence that any difference exists between trained and novice readers' cognitive processes on how they understand UML class diagrams. If evidence exists that there are differences in cognitive processes, then the results of study 1 can be strengthened. Therefore, the objectives of this follow-up protocol study are to: (1) identify the difference in mental processes of trained and novice readers in understanding the models and (2) understand whether the differences support the hypotheses.

In IS research, verbal protocol analysis is used to understand the processes by which users perform tasks. This technique provides data that reveal the mental processes that take place as individuals work on problem-solving tasks (Ericsson & Simon, 1984). To use this technique, the subjects are required to verbalize their thought processes and strategies as they solve specific problem-solving tasks.

#### ***Experimental Design***

To limit the scope of the study, we provide only weak construct redundant models to a small sample of ten subjects (five trained and five novices). Consistent with other such studies (e.g., (Burton-Jones & Meso, 2006)), this number of subjects provides immense verbal data for such a small sample size. The subject's profile, experimental procedure, tasks were similar to that of the earlier study, except that subjects had to verbalize their thought processes while performing the problem-solving tasks. A different set of subjects was selected in this study.

### ***Results***

The verbal data was transcribed and coded by a PhD student in conceptual modeling who was unaware of the objectives of the study. The analysis was done by collecting evidence on the behavior of readers while answering the problem-solving tasks. The analysis is reported in Tables 7 and 8.

Earlier studies have demonstrated that experts depend on long-term memory whereas novices depend on short-term memory for solving problems (Glaser, 1984). Thus, trained readers should use their long-term memory to understand the model concepts whereas novices should rely on the diagrams. The trained readers used words or phrases that refer to the constructs in the diagrams much more than the novices (Tables 7, 8). For example, the trained readers used words such as generalization, composition, aggregation, "sub class of," and "consists of" to refer to the labels "isPartOf", "isA", and "isComponentOf". These words were not present in the models. On the other hand, as expected, the novices read the labels directly from the diagrams to understand the model concepts.

Previous conceptual modeling studies find that trained readers understand the model better than novices (Batra & Davis, 1992). In this study, we observe that trained readers were able to make more inferences from the diagrams than novices (Tables 7, 8). Some examples of

trained readers' verbalization that reflect inferences include “exhibited animals have same attributes as species” (inferred from ExhibitedAnimal is a subclass of ShipmentAnimal which is a subclass of Species) and “though crew do not consist of pilots” and “flight engineer are all part of the crew but not pilot...” (both phrases were inferred from the classes that are related to the crew, only pilots had association with the crew and other classes had aggregation relations with the crew). Also, only trained readers demonstrated that they could read the relations between the classes from both directions especially when the aggregation, generalization, and composition labels were used. For example, trained readers verbalized “there are two types of shipment animals” and “aircraft consists of cargo.” In contrast, novices read the diagrams in one direction only using the labels (e.g., cargo is part of the aircraft). These observations reflect that unlike novices, trained readers relied more on their long-term memory than short-term memory, and trained readers understood the diagrams better than novices.

Table 7: Analysis of verbal data- trained readers

Trained readers	Verbalizing concepts not in the diagram	Verbalizing labels in the diagram	Number of times reading reverse/ inference
<b>Flight domain</b>			
1	3	1	2
2	10	1	1
3	2	2	2
4	1	0	0
5	2	2	2
Subtotal	<b>18</b>	<b>6</b>	<b>7</b>
<b>Aquarium management domain</b>			
1	4	1	0
2	6	3	2
3	1	1	0
4	2	2	1
5	1	0	2
Subtotal	<b>14</b>	<b>7</b>	<b>5</b>
<b>Total</b>	<b>32</b>	<b>13</b>	<b>12</b>

Table 8: Analysis of verbal data-novice readers

Novice readers	Verbalizing concepts not in the diagram	Verbalizing labels in the diagram	Number of times reading reverse/inference
<b>Flight domain</b>			
1	0	1	0
2	1	1	1
3	0	1	0
4	1	1	1
5	0	0	0
Subtotal	<b>2</b>	<b>4</b>	<b>2</b>
<b>Aquarium management domain</b>			
1	0	2	0
2	0	2	0
3	2	4	0
4	0	1	0
5	0	3	0
Subtotal	<b>2</b>	<b>12</b>	<b>0</b>
<b>Total</b>	<b>4</b>	<b>16</b>	<b>2</b>

The above results should be used with some caution as the protocol analysis has a limitation in that the subjects might not explicitly verbalize their thought processes (Dimoka, Pavlou, & Davis, 2011). Thus, the actual number of instances as indicated in Tables 7 and 8 cannot be determined accurately. However, the results can still be used to understand the differences in the cognitive processes of the two groups. While the novices rely on the labels as depicted in the diagrams, the trained readers rely more on their long-term memory. Thus the results support study 1. Although the labels are redundant, the trained readers do not seem to use them to understand the diagrams and thus they do not face any redundancy effect. Therefore, for the trained readers there is not much difference between ontologically clear and redundant models. In contrast, novices primarily used the labels to understand the new constructs. Thus, for them the redundant models are more useful than the clear models.



#### 4. Discussion

Predictions based on ontological analysis of conceptual modeling grammars using the Bunge-Wand-Weber representation model as reference ontology have mostly been supported by empirical testing (Wand & Weber, 2017). A notable exception is the predicted detrimental effect of ambiguity caused by construct redundancy, which was only investigated for process modeling and with mixed effects (Fickinger & Recker, 2013). Our research results provide further insights into when and why construct redundancy in conceptual modeling grammars has positive or negative effects on model understanding.

To perform our research, we operationalized construct redundancy by means of a conceptual modelling grammar based on UML class diagrams. First, we made a distinction between two types of redundancy in conceptual models. A weak form of redundancy in models is introduced by extending a symbolic grammatical construct with a textual label that acts as secondary notation (Petre, 2006), hence adhering to the dual code principle (Moody, 2009). Our first empirical study showed that consistently using this grammatical construct with redundant coding throughout the model for representing the same ontological construct significantly improved model understanding of novice users as measured by their performance of problem-solving tasks. This positive effect was predicted by the complementarity of the redundancy effect for readers that are not completely familiar with the symbols used for encoding the information in the diagram (Ainsworth, 2006). Our second study confirmed this prediction by means of a protocol analysis study which showed that such novice readers rely much more on the textual information (i.e., labels) in a domain model than more advanced readers. This is in line with the study of Recker et al. (2010) who find that the reduced perceived ease of use of BPMN because of its construct redundancy for representing real-world objects and events holds for those users

who can identify these ontological deficiencies. Clearly, for novice users the labels added to the graphical symbols are not perceived as redundant and actually help them understand the information conveyed by the model. As expected, this positive effect did not hold for the more knowledgeable users who were neither helped nor distracted by the redundant information. Our protocol analysis confirmed this.

We also introduced a stronger form of construct redundancy in models by creating three alternative representations of the same ontological construct. Redundancy was induced into the models by having the same ontological construct represented in different ways in the same model, resulting in non-laconic models (Guizzardi, 2005). We predicted that the ambiguity caused by such alternative representations is likely to cause increased cognitive load, especially for novice users. This prediction was confirmed by the results of our first study which showed a significant decrease of problem-solving performance for novice users when using strong construct redundant models compared to ontologically clear models. Contrary to our expectations, also the more advanced users saw their performance decrease, though the effect size was low. Previous studies by Recker and Rosemann (2010) and Recker et al. (2010) concluded that experienced modelers perceive ontological deficiencies of conceptual modeling grammars and also find ways to mitigate the deficiencies. These studies were survey-based and did not directly investigate performance effects caused by the ambiguity of ontologically unclear grammars. In contrast, our empirical study demonstrated such negative effect of strong redundancy in conceptual models, both for novice and more advanced users, hence providing further evidence for Wand and Weber's theory of ontological expressiveness.

#### **4.1 Implications for Research and Practice**

Our studies suggest two implications for research. First, in further research on theorizing, applying, and testing ontological predictions (see Wand and Weber (2017) for possible future research avenues on the theory of ontological expressiveness), construct redundancy needs to be carefully conceptualized and operationalized. Our research demonstrates differential effects for deliberate kinds of redundancy (i.e., secondary notation, dual coding) and unwanted construct redundancy that is built in the grammar's design. Further, even when a grammar offers multiple constructs to represent the same ontological construct, models can be constructed that are laconic (Guizzardi, 2005), meaning that only one grammatical alternative is actually used to represent the ontological construct in the models. The latter consideration is especially relevant for studies investigating the impact of lack of ontological clarity caused by construct redundancy in a grammar on model understanding.

Second, ontological predictions might need to be differentiated for separate groups of target users of the grammar and/or scripts. Our second study shows that model readers that are new to the grammar and more knowledgeable with the grammar use different cognitive processes when understanding models, and our first study demonstrates different outcomes for those two groups. This means that contingency factors need to be accounted for when investigating the theory of ontological expressiveness. Apart from experience and knowledge, other human aspects of conceptual modeling (Topi & Ramesh, 2002) such as domain familiarity can be considered. Suh and Park (2017) investigate how domain familiarity affects the user's performance in cases of construct overload. Similarly, future studies can investigate the role of domain familiarity in construct redundancy. This research can also be replicated with expert model readers (e.g., practitioners) and can be extended by studying the effect of a large variety of redundant constructs in conceptual models.

Our research also suggests practical implications. It shows the advantages of secondary notation in conceptual models. Providing textual labels as additional clues improves understanding of users who are not familiar with the modelling grammars. On the other hand, distinct construct redundancy is to be avoided for all model users. In case modelling grammars are used that exhibit such ontological deficiency (e.g., BPMN (Recker et al., 2010), UML (Opdahl & Henderson-Sellers, 2002)), modelers should be discouraged using multiple grammatical constructs in representing ontological constructs. Proper rules or guidelines should be developed and used to represent ontological constructs with minimum grammatical constructs, in order to create laconic models that do not suffer from strong detrimental forms of redundancy.

## **4.2 Limitations**

The results of the studies depend on the operationalization of the grammatical constructs and manipulation of modeling knowledge. For example, in the weak redundant model without a symbol and a label as a generalization construct, if a new grammatical construct (e.g., a curved line) is used, then the results could have been different. The weak and strong redundant models were operationalized, keeping in mind the subjects who were recruited in this study. If novice readers are quite familiar with the ontological constructs then these readers may not find difficulty in understanding the semantics of the redundant grammatical constructs.

A key weakness of this study is the use of students as subjects. The amount of modeling exposure that students get is limited in academic programs, and thus the results can vary when students with different background are used.

## 5. Conclusion

This paper investigates the role of redundancy in understanding conceptual models by readers with varied modeling knowledge. The main contribution of this study is the role of redundancy in UML class diagrams being investigated at the readers' level. This research might explain the different results obtained in the past on investigating the role of construct redundancy. If trained model readers are used in studies, then there should not be any effect of redundancy on task performance. But if novice readers are used, then some effect from redundancy exists. Thus, depending on the readers' knowledge of models, the effect of redundancy is mixed.

There are several ways to extend this research. Ali Jabbari Sabegh and Recker (2017) mention that most studies that investigate ontological deficiencies focus on a single conceptual modeling grammar or model. But as multiple conceptual models are generally used simultaneously in practice, therefore the role of redundancy can be investigated in the context of using multiple conceptual models. Kaul, Storey, and Woo (2017) suggest a framework on guiding complexity of information system design. Using Wand and Weber's decomposition model (Wand & Weber, 1990), they suggest how complexity can be reduced by decomposing information systems. Research can focus on how removal of construct redundancy can help in decomposition of conceptual models and subsequently reduce complexity of information systems.

## Appendix A: Experimental Diagrams: Conceptual models

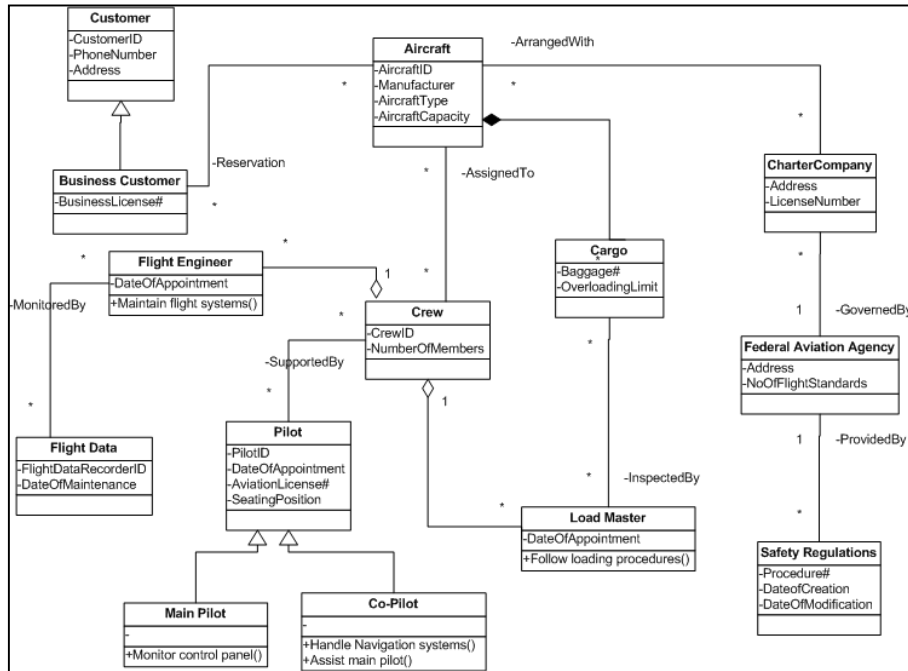


Figure A1: Ontologically clear model- Flight domain

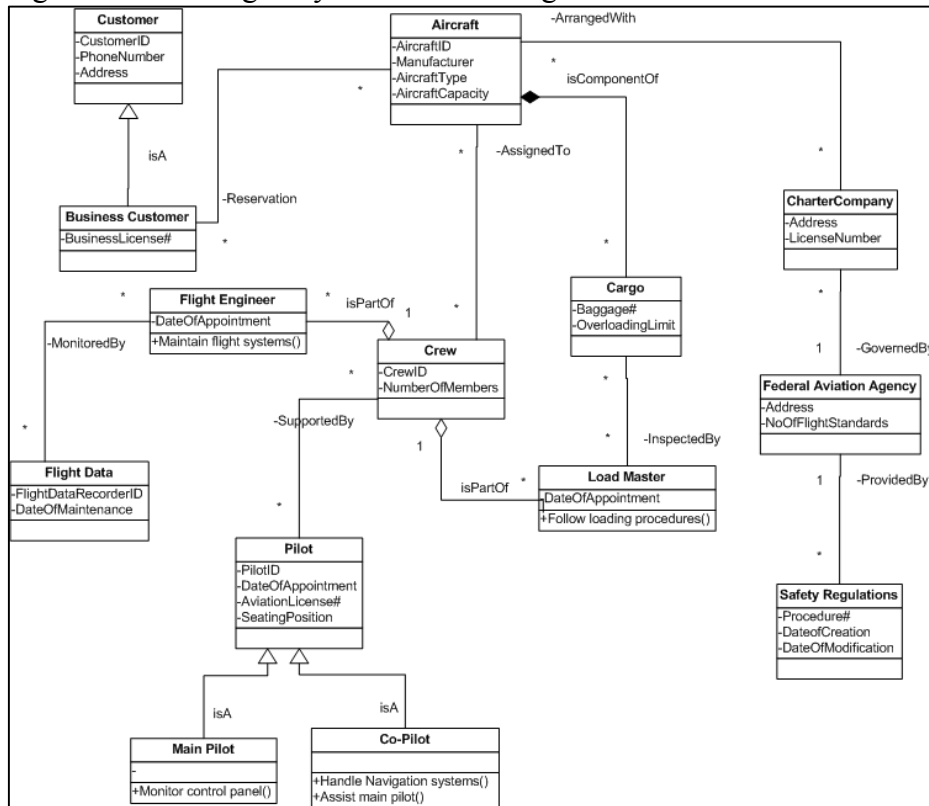


Figure A2: Weak redundant model- Flight domain

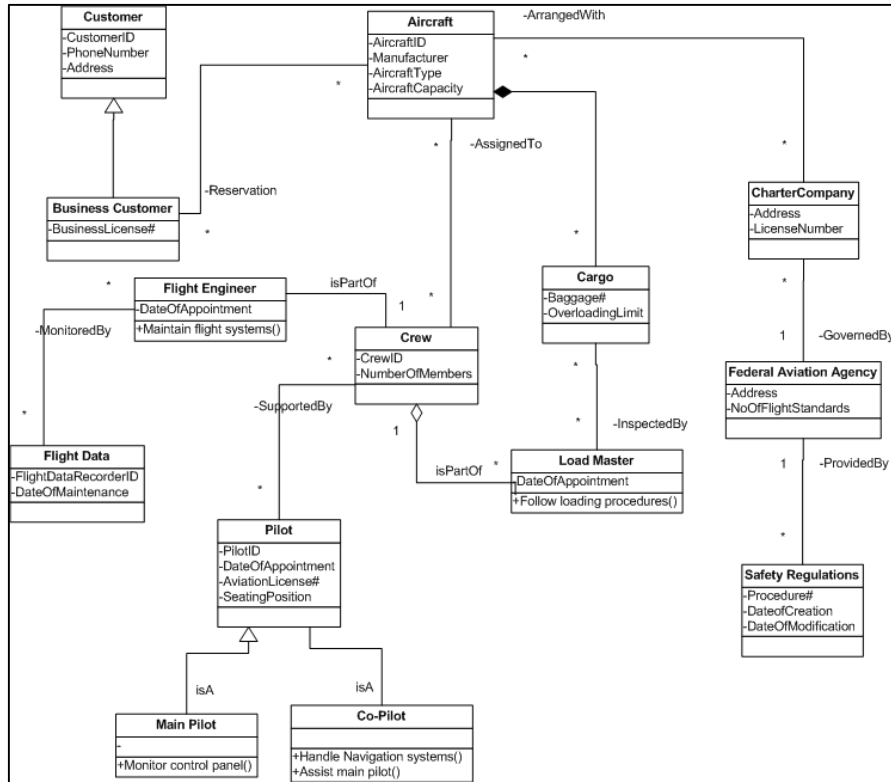


Figure A3: Strong redundant model-Flight domain

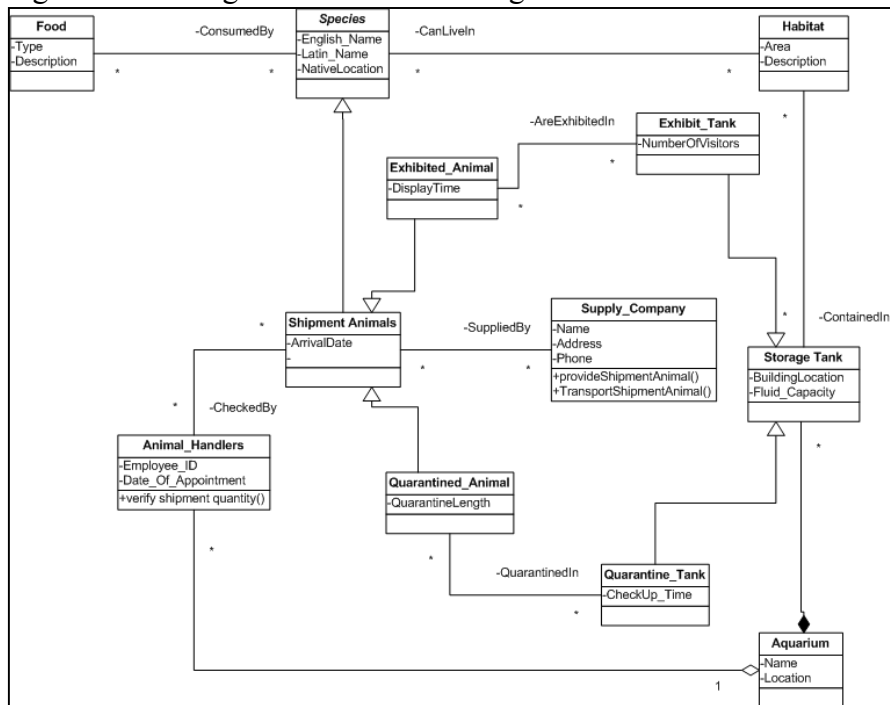


Figure A4: Ontologically clear model- aquarium management domain

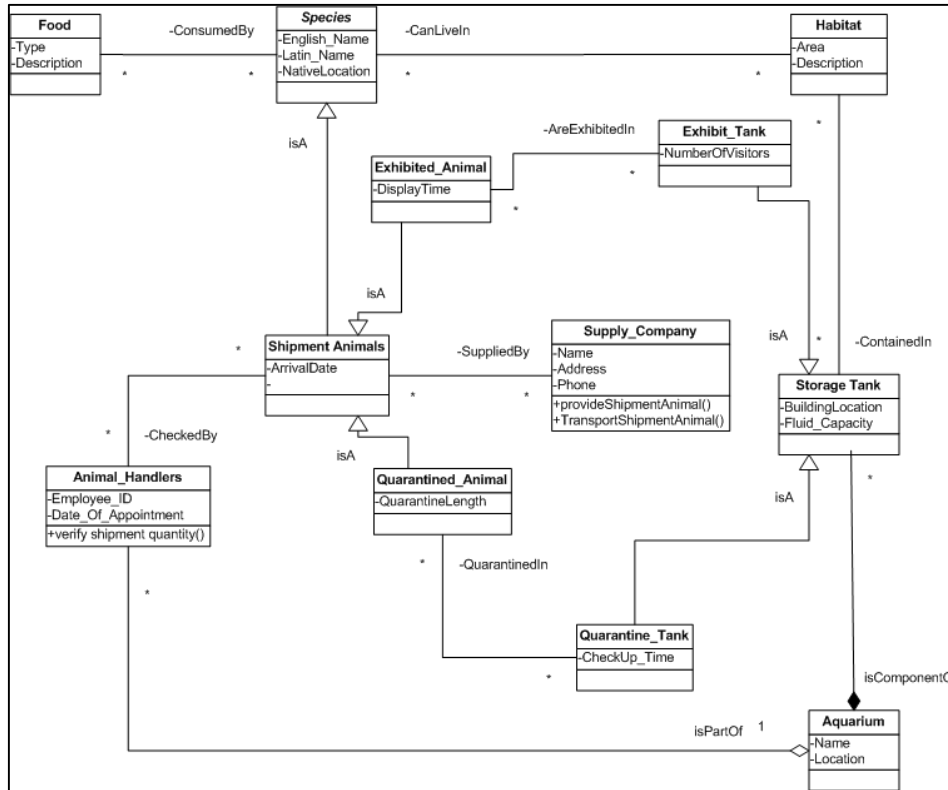


Figure A5: Weak redundant model- aquarium management domain

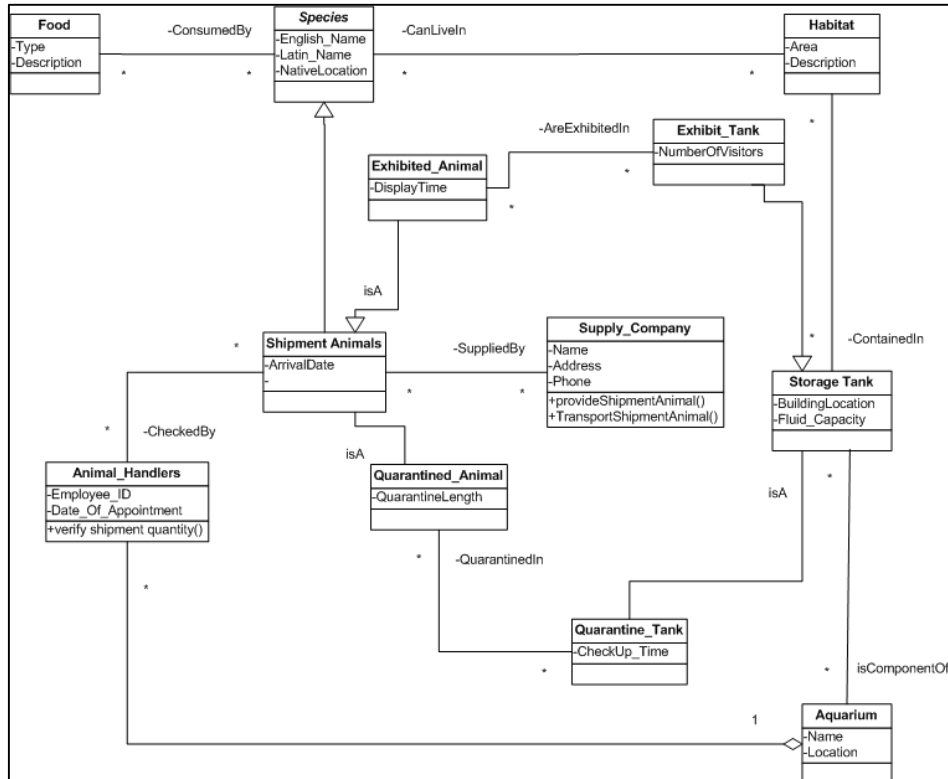


Figure A6: Strong redundant model- aquarium management domain



## Appendix B: Variables

### Variables: Modeling and domain knowledge (on a scale of 1-7)

1. To what extent do you know data modeling concepts (such as classes, operations, and attributes)
2. To what extent do you have experience in using data modeling concepts (such as classes, operations, and attributes)
3. To what extent do you know aggregation, composition, and generalization
4. To what extent do you have experience in using aggregation, composition, and generalization
5. To what extent do you know the operation of a chartered flight?
6. To what extent are you familiar with the operation of a chartered flight?
7. To what extent are you familiar with the different types of employees that work in an Aquarium?
8. To what extent do you know the responsibilities of employees working in an Aquarium?

### Dependent Variable: Problem-solving questions (open-ended)

1. Certain pilots were on strike due to low wages. How can a chartered company still operate chartered flights without affecting the operations?
2. Some of the safety procedures were not followed by a few crew members during a certain chartered flight? Explain how this could have happened?
3. The cargo of a flight was overloaded with more than 10% of its allowed weight? Explain how this could have happened?
4. A new species of animal has just been discovered in Tasmania. The aquarium is thrilled to purchase this species. What measures should the aquarium take before it is displayed to the visitors?
5. Certain species of animals do not feed for few days when they are brought in the aquarium. What measures will you suggest to make sure that these animals remain healthy?
6. An animal from this aquarium needs to be transferred to another aquarium located in a different country. What steps should be taken to transfer the animal?

### Comprehension questions (True/False)

1. A flight engineer is a crew
2. A load master is a part of a crew
3. A pilot is a part of a crew
4. A crew can be assigned to more than one aircraft
5. A cargo is a component of aircraft
6. A flight engineer inspects loading
7. A flight engineer monitors flight data
8. A co-pilot has an aviation license number
9. A load master has an attribute CrewID
10. An animal handler is a part of an aquarium
11. All shipment animals are quarantined
12. All shipment animals are exhibited
13. Exhibit Tank has the attribute Fluid\_Capacity
14. A storage tank is a component of an aquarium
15. Quarantine Tank has the attribute Fluid\_Capacity
16. All shipment animals are species
17. Shipment\_Animals has an attribute Native\_Location
18. Exhibited\_Animals has an attribute Arrival\_Date

### Task on developing UML class diagram

In the following space draw a UML class diagram for the description below using at least three classes.  
“A hospital treats patients. Some of the treated patients are admitted to the hospital.”

## References

- Ainsworth, S. E. (2006). DeFT: A conceptual framework for learning with multiple representations *Learning and Instruction*, 16(3), 183-198.
- Allen, G. N., & March, S. T. (2006). The Effects of State-Based and Event-Based Data Representations on User Performance in Query Formulation Tasks. *MIS Quarterly*, 30(2), 269-290.
- Batra, D., & Davis, J. (1992). Conceptual Conceptual modeling in Database Design: Similarities and Differences between Novices and Expert Designers. *International Journal of Man-Machine Studies*, 37, 83-101.
- Bodart, F., Patel, A., Sim, A., & Weber, R. (2004). Should Optional Properties be Used in Conceptual Modeling? A Theory and Three Empirical Tests. *Information Systems Research*, 12(4), 383-405.
- Bowen, P. L., O'Farrell, R. A., & Rohde, F. H. (2009). An Empirical Investigation of End User Query Development: The Effects of Improved Model Expressiveness Versus Complexity. *Information Systems Research*, 20(4), 565-584.
- Burton-Jones, A., & Meso, P. (2006). Conceptualizing Systems for Understanding: An Empirical Test of Decomposition Principles in Object-Oriented Analysis. *Information Systems Research*, 17(1), 38-60.
- Chandler, P., & Sweller, P. (1991). Cognitive Load Theory and the Format of Instruction. *Cognition and Instructions*, 8(4), 293-332.
- Dimoka, A., Pavlou, P., & Davis, F. D. (2011). NeuroIS: The Potential of Cognitive Neuroscience for Information Systems Research. *Information Systems Research*, 22(4), 687-702.
- Dobing, B., & Parsons, J. (2006). How UML is used. *Commun. ACM*, 49(5), 109-113. doi:10.1145/1125944.1125949
- Dunn, C., Gerard, G. J., & Grabski, S. (2011). Diagrammatic Attention Management and the Effect of Conceptual Model Structure on Cardinality Validation. *Journal of the Association for Information Systems*, 12(8), 586-604.
- Ericsson, K. A., & Simon, H. A. (1984). *Protocol Analysis: Verbal Reports as Data*. Cambridge, Mass.: MIT Press.
- Fickinger, T., & Recker, J. (2013). *Construct Redundancy in Process Modeling Grammars: Improving the Explanatory Power of Ontological Analysis*. Paper presented at the 21st European Conference on Information Systems, Utrecht.
- Gemino, A. (1998). *Comparing Object Oriented with Structured Analysis Techniques in Conceptual Modeling (PhD Thesis)*. University of British Columbia, Vancouver.
- Gemino, A., & Wand, Y. (2005). Complexity and clarity in conceptual modeling: Comparison of mandatory and optional properties. *Data and knowledge Engineering*, 55, 301-326.
- Glaser, R. (1984). Education and thinking: The role of knowledge. *American Psychologist*, 39(2), 93-104.
- Green, P., & Rosemann, M. (2001). Ontological Analysis of Integrated Process Models: Testing Hypotheses. *Australasian Journal of Information Systems*, 9(1), 30-38.
- Guizzardi, G. (2005). *Ontological Foundations for Structural Conceptual Models*. (PhD), University of Twente, The Netherlands.
- Jabbari Sabegh, M., & Recker, J. (2017). Combined Use of Conceptual Models in Practice: An Exploratory Study. *Journal of Database Management*, 28(2).

- Kalyuga, S., Ayres, P., Chandler, P., & Sweller, J. (2003). The Expertise Reversal Effect. *Educational Psychologist*, 38(1), 23-31.
- Kaul, M., Storey, V., & Woo, C. C. (2017). A Framework for Managing Complexity in Information Systems. *Journal of Database Management*, 28(1).
- Khatri, V., Vessey, I., Ramesh, V., Clay, P., & Park, S. (2006). Understanding Conceptual Schemas: Exploring the Role of Application and IS Domain Knowledge. *Information Systems Research*, 17(1), 81-99.
- Mayer, R. E. (1989). Models for Understanding. *Review of Educational Research*, 59(1), 43-64.
- Moody, D. (2009). The Physics of Notations: Toward a Scientific Basis for Constructing Visual Notations in Software Engineering. *IEEE Transactions on Software Engineering* 35(6), 756 - 779.
- Mylopoulos, J. (1992). Conceptual modeling and telos. In P. a. Z. Locuopoulos, R. (Ed.), *Conceptual modeling, Databases and Cases*. New York: John Wiley and Sons Inc.
- Opdahl, A., & Henderson-Sellers, B. (2002). Ontological evaluation of the UML using the Bunge-Wand-Weber model. *Software and systems modeling*, 1(1), 43-67.
- Petre, M. (2006). Cognitive dimensions 'beyond the notation'. *Journal of Visual Languages & Computing*, 17(4), 292-301.
- Polya, G. (1968). *Mathematical discovery: on understanding, learning and teaching problem solving* (Vol. II). New York: Wiley.
- Recker, J., Indulska, M., Rosemann, M., & Green, P. (2010). The Ontological Deficiencies of Process Modeling in Practice. *European Journal of Information Systems*, 19(5), 201-525.
- Recker, J., & Rosemann, M. (2010). A Measurement Instrument for Process Modeling Research: Development, test and procedural model. *Scandinavian Journal of Information Systems*, 22(2).
- Robertson, L. C. (2003). *Space, Objects, Minds and Brains*. New York: Psychology Press.
- Shanks, G. (1997). Conceptual Conceptual Modeling: An Empirical Study on Expert and Novice Conceptual Modelers. *Australian Journal of Information Systems*, 4(2), 63-73.
- Shanks, G., Tansley, E., Nuredini, J., Tobin, D., & Weber, R. (2008). Representing Part-Whole Relations in Conceptual Modeling: An Empirical Evaluation. *MIS Quarterly*, 32(3), 553-573.
- Suh, J., & Park, J. (2017). Effects of Domain Familiarity on Conceptual Modeling Performance. *Journal of Database Management*, 28(2).
- Sweller, J. (1988). Cognitive Load During Problem Solving: Effects on Learning. *Cognitive Science*, 12(2), 257-285.
- Tilakaratna, P., & Rajapakse, J. (2017). Evaluation of the Ontological Completeness and Clarity of Object-Oriented Conceptual Modelling Grammars. *Journal of Database Management*, 28(2).
- Topi, H., & Ramesh, V. (2002). Human factors research on data modeling: A review of prior research, an extended framework and future research directions. *Journal of Database Management*, 13(2).
- Wand, Y., & Weber, R. (1990). An Ontological Model of an Information System. *IEEE Transactions on Software Engineering*, 16(11), 1282-1992.
- Wand, Y., & Weber, R. (1993). On the Ontological Expressiveness of Information Systems Analysis and Design Grammars. *Journal of Information Systems*, 3, 217-237.
- Wand, Y., & Weber, R. (2017). Thirty Years Later: Some Reflections on Ontological Analysis in Conceptual Modeling. *Journal of Database Management*, 28(1).

- Ward, M., & Sweller, J. (1990). Structuring Effective Worked Examples. *Cognition and Instructions*, 7(1), 1-39.
- Weber, R. (2003). Conceptual Modelling and Ontology: Possibilities and Pitfalls. *Journal of Database Management*, 14(3), 1-20.
- Weber, R., & Zhang, Y. (1996). An Analytic Evaluation of NIAM's grammar for Conceptual Schema Diagrams. *Information Systems*, 6(2), 147-170.

### **Author Bio**

Palash Bera

Palash Bera is an associate professor in Operations and IT Management department at the John Cook School of Business, Saint Louis University, St. Louis. His research interests are in empirical studies of information systems analysis and design and the use of eye tracking technologies as a research methodology. His research focuses on creating and testing guidelines of structured requirements analysis. His recent research interests focus on combining requirements analysis with software testing. In particular, developing guidelines that can help software test design. He has published in reputed information systems journals such as MIS Quarterly, Information Systems Research, and Communications of the ACM. He has worked in executive and consulting positions in two multinational companies. His teaching interests are in Business Analytics, Requirements Analysis, and Database Design. He has recently served as the President of the AIS Special Interest Group on Systems Analysis and Design.

Geert Poels

Geert Poels is a full professor of Management Information Systems and member of the professorial staff of the Faculty of Economics and Business Administration, Ghent University, Belgium. As head of the Business Informatics research team he directs research on the modelling and analysis of strategic fit, value delivery, business capabilities, and business services. In the period 2008-2016 he supervised 10 completed PhD research projects at Ghent University and KULeuven, while currently he is a promoter of 8 ongoing PhD projects. His personal research interests concern the quality of conceptual models, the conceptual modelling of service systems, business process architecture, and the disruptive impact of information technology on business models and business processes. He published in Information Systems Journal, European Journal of Information Systems, Journal of Database Management, Decision Support Systems, IEEE Transactions on Services Computing, IEEE Transactions on Software Engineering, Expert Systems with Applications, and other academic journals.